CLOUD BASED ANOMALY DETECTION IN COMPUTER NETWORKS USING MACHINE LEARNING TECHNIQUES

Ram Paul Hathwal
Dept. of CSE
Amity School of Engineering and Technology, Delhi, India
rp.hathwal@yahoo.com

Rishabh Jain
Dept. of CSE
Amity School of Engineering and Technology, Delhi, India
rishabhjain.1936@gmail.com

Bhavay Anand
Dept. of CSE
Amity School of Engineering and Technology, Delhi, India
bhavayanand9@gmail.com

Nitin Verma
Dept. of CSE
Amity School of Engineering and Technology, Delhi, India
vermanitin1998@gmail.com

Prabodh Ranjan Swain
Dept. of CSE
Amity School of Engineering and Technology, Delhi, India
prabodhranjan98@gmail.com

***Abstract*- With increasing number of cyber attacks on organization, cyber security now plays a crucial role in protecting the organizations assets. These attacks range from simple attacks such as portscan which is used in footprinting and reconnaissance stage to Denial of Service (DoS) attacks which can tamper with the services offered by the organization such as web services and many more. Other attacks which are very prominent include SQL injection, cross-site scripting (XSS) or some passwords attacks like brute forcing and many more. The organizations deploy various mechanisms to protect their assets from such attacks. These mechanisms include deployment of honeypots, firewalls, Intrusion Detection Systtems (IDS) and Intrusion Prevention Systems (IPS). In this paper we design a behavior-based IDS which can detect these attacks. The IDS use machine learning techniques to learn about various attacks and can then detect attacks. The advantage of such IDS is that it can also detect various zero-day attacks based on similarity with past attacks.**

Keywords- Cyber Security, Machine Learning, Cloud, KNN, DoS, DDoS, PortScan, Web Attacks, IDS

I. INTRODUCTION

At present time everything is connected to the Internet. The Internet is a collection of various computational devices which are able to participate in the process of information generation as well as information sharing. These devices can be portable such as laptops and cell phones or fixed such as big data centers. As the world is expanding at an enormous rate, Internet is being used in almost every sector such as healthcare, defence and military, social media, entertainment and many more. In order to grab attention of the public and encourage people to use the services of a particular company, many multi-national corporations try to introduce new technologies like fog computing, grid computing, mobile computing and cloud computing [1].

With the increase in the usage of Internet a new problem arose i.e. Cyber Security.

An anomaly can be defined as deviation from the norm or median. The anomaly can deviate either positively or negatively. When the deviation is negative, the chances of a threat effecting our system increases. The aim of cyber security is to maintain the Confidentiality, Integrity and Availability of the data stored in the computational devices or the data that is being generated and shared between two or more devices.

Some of the attacks that happen on the Internet include Denial of Service attack, SQL injection, Cross Site Scripting (XSS), Brute Forcing, Cross Site Request Forgery (CSRF) and many more.

In order to prevent from these attacks many corporations use Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS). There are different types of IDS available [2]:

1. Signature based IDS – detect attack based upon pre-configured knowledge base. This type of IDS has high detection accuracy if the attack is previously known and its signature is present in the knowledge base.
2. Anomaly Detection IDS – Statistical and collective behavior is used. This lowers the false alarm rate and detection of new attacks is made easier.
3. Host based IDS (HIDS) – it is deployed on the host. It monitors the host files, system calls and network events. The main disadvantage is that this type of IDS is not suitable for network-based threats.
4. Network based IDS (NIDS) – this IDS monitors the network traffic. Encrypted traffic and the attack which is performed during the high flow rate of the packets in the network reduces the efficiency of NIDS. However, the NIDS is able to monitor multiple at a time.
5. Distributed IDS (DIDS) – this IDS is a combination of HIDS and NIDS. However, the operational cost and communicational cost is high.

Machine learning approach can be used to improve the efficiency of the IDS. Machine learning consists of various algorithms which can be used. These algorithms can be categorized as Supervised learning, Unsupervised learning and evolutionary learning.[3]

1. Supervised learning – in this form of learning the dataset is readily available and the training dataset points to the target vector. There are many algorithms in supervised learning. These are:
   - 1.1 Decision Tree
   - 1.2 Naïve Bayes
   - 1.3 Bayesian Network
   - 1.4 Logistic Regression
   - 1.5 Neural Networks
   - 1.6 Support Vector Machines

2. Unsupervised learning – in this form of learning the machine learns from the data and the environment. There is no availability of target vector. The algorithms involved are:

2.1 K-Means
2.2 CLIQUE

3. Evolutionary learning – the evolutionary learning broadly consists of 4 types of algorithms:

3.1 Genetic Algorithm
3.2 Particle Swarm Optimization
3.3 Ant Colony Optimization
3.4 Artificial Immune System

Feature transformation, feature reduction and feature selection are three techniques involved in machine learning approach. Feature transformation involves the processing of the information. The information is converted from one form to another in order to reduce the computational overhead.
Feature reduction means reduction of dimensions of the selected features. Bigger the dimensions of the features more will be the complexity.
Feature selection involves the selection of the required features from the superset of the features. This process is necessary to reduce irrelevant features.

II. METHODOLOGY

A. Software Platform

- **Python** - a free and open source object-oriented programming language, draws attention with its simple syntax and dynamic structure. In Python, it's very easy to write code and analyse code. Another advantage is that it has the advantage of extensive documentation (books, internet sites, forums, etc.). In addition to all these advantages, it works in concert with many libraries which "machine learning" applications can be done. In this context, Python3.6 has been chosen to be used in this work, because of many of the advantages it provides.
- **Sklearn** - (Scikit-learn) is a machine learning library that can be used with the Python programming language. Sklearn offers a wide range of options to the user with its numerous machine learning algorithms. Sklearn has extensive documentation and contains all the algorithms needed for this work.
- **Pandas** - a powerful data analysis library running on Python. When working with a large dataset, Pandas allows you to easily perform many operations such as filtering, bulk column / row deletion, addition, and replacement. Because of all these advantages, the Pandas library has been used.

B. Performance Evaluation Methods

The results of this study are evaluated according to four criteria, namely accuracy, precision, f-measure, and recall. All these criteria take a value between 0 and 1. When it approaches 1, the performance increases, while when it approaches 0, it decreases.

Accuracy: The ratio of successfully categorized data to total data

$$\text{Accuracy} = \frac{TN+TP}{FP+TN+TP+FN} \qquad (1)$$

Precision: The ratio of successful classified data as the attack to all data classified as the attack

$$\text{Precision} = \frac{TP}{FP+TP} \qquad (2)$$

F-measure (F-score/F1-score): The harmonic-mean of sensitivity and precision. This concept is used to express the overall success. So, in this study, when analysing the results, it will be focused, especially on the F1 Score.

$$\text{F-measure} = \frac{2}{\frac{1}{Recall}+\frac{1}{Precision}} \qquad (3)$$

C. Implementation

In this section, various pre-processing and actual application are performed to detect anomaly by machine learning techniques. For this purpose, the data cleansing process is performed in the first step and the dataset is cleaned from mistakes and defects. Then, the data set is divided into two parts, training, and test. After these operations, the properties to be used by the algorithms are decided at the step of feature selection. Finally, the section ends with the implementation of machine learning algorithms.

D. Data Cleansing

It may be necessary to make some changes to the dataset before using it in practice, making it more efficient. For this purpose, in this section, some defects of the CICIDS2017 dataset are corrected, and some data are edited.

When the records in the dataset are examined, it can be seen that the many record are incorrect / incomplete. The first step in the pre-processing process will be to delete these unnecessary records.
Another error about the dataset is in the columns that make up the features. The dataset file consists of 86 columns that define the flow properties such as Flow ID, Source IP, Source Port etc. However, the Fwd Header Length feature (which defines the forward direction data flow for total bytes used) was written two times (41st and 62nd columns). This error is corrected by deleting the repeating column (column 62).

Another change that needs to be made in the dataset is to convert the properties including the categorical and string values (Flow ID, Source IP, Destination IP, Timestamp, External IP) into numerical data to be used in machine learning algorithms. This can be done with *LabelEncoder* from *Sklearn* classes. In this way, various string values that cannot be used in machine learning operations will get integer values between 0 and n-1 and will become more suitable for processing.

However, although the “Label” tag is a categorical feature, no changes have been made on it. The reason is that during the

processing, the original categories are needed in order to classify the attack types in different forms and to try different approaches.
Finally, some minor structural changes should be made to the dataset, including:

- In the Label feature, the character "–" (Unicode Decimal Code &#8211) used to identify the web attack subtypes (Web Attack - Brute Force, Web Attack - XSS, Web Attack - SQL Injection) must be replaced with the character "-" (Unicode Decimal Code &#45), since utf-8, the default codec of Pandas library, does not recognize it. Otherwise, the Pandas library that will not recognize this character and it will fail.

- "Flow Bytes/s", "Flow Packets/s" features include the values "Infinity" and "NaN" in addition to the numerical values, which can be modified to -1 and 0 respectively to make them suitable for machine learning algorithms.

E. Creation of Training and Test Data

During the machine learning process, data is needed so that learning can take place. The data sets used are the result of this need. In addition to the data required for training, test data is needed to evaluate the performance of the algorithm and to see how well it works. The algorithm acquires a skill on the training data and applies it to the test data. The result of the test data is the performance of the machine learning algorithm.

However, the CICIDS2017 dataset used does not contain dedicated training and test data, but it contains a single unbundled dataset. Therefore, the data should be divided into training and test data parts. In the application phase, a *Sklearn* command, *train_test_split* is used. This command divides the data into 2 parts at the sizes specified by the user. Generally preferred partitioning is 20% test, 80% training data and this ratio is also preferred in this application. The *train_test_split* command makes the selection random when creating data groups. This process is known as cross-validation. In order to ensure that the results obtained during the application are solid, the creation of the training and the test data have been performed 10 times in succession. The results obtained are the arithmetic mean of the repeated operations.

F. KNN for classification

KNN (K Nearest Neighbour), which is a sample-based method, is one of the most used machine learning algorithms with its simple and fast structure. This algorithm depends on the assumption that the examples in a dataset will exist close to the examples with similar properties in another dataset.

In this context, KNN identifies the class of new data that is not classifiable by using training data of known class type. This determination is made by observing the nearest neighbours of the new sample, for which no classifications are specified.

In a plane with N properties, the number of neighbours to be looked at for an unclassified sample is specified by the number K. For the unknown sample, the distances to the neighbours are calculated and the smallest K numbers are chosen from these distance values. The most repeated property within the K values is assigned as the unknown instance property.

KNN, which provides good performance over multidimensional data and is a fast algorithm during the training phase, is relatively slow in the estimation stage

In pattern recognition, the KNN algorithm is a method for classifying objects based on closest training examples in the feature space. KNN is a type of instance-based learning, or lazy learning where the function is only approximated locally and all computation is deferred until classification. The KNN is the fundamental and simplest classification technique when there is little or no prior knowledge about the distribution of the data. This rule simply retains the entire training set during learning and assigns to each query a class represented by the majority label of its k-nearest neighbours in the training set.

The Nearest Neighbour rule (NN) is the simplest form of KNN when K = 1. In this method each sample should be classified similarly to its surrounding samples. Therefore, if the classification of a sample is unknown, then it could be predicted by considering the classification of its nearest neighbour samples. Given an unknown sample and a training set, all the distances between the unknown sample and all the samples in the training set can be computed. The distance with the smallest value corresponds to the sample in the training set closest to the unknown sample. Therefore, the unknown sample may be classified based on the classification of this nearest neighbour.

III. RESULTS

When looking at the results, it is noticed that KNN performs reasonably well on a wide variety attack detection with high F-measure scores. It is well known that KNN gives high accuracy predictions on classification problems with low dimensionality training vector space. We can compare figures generated from both algorithms i.e. KNN and K-Means in terms of F-measure score and compare their relative performance when using different parameter values.

In the implementation we have tested our model against to machine learning algorithms i.e. KNN and K-Means algorithm. Table 3 and Figure 1, Figure 2 and Figure 3 show the comparison of the results obtained from two studies. When the results obtained in them are examined, the results given by the KNN Algorithm fits best at the neighbour size of 5.

It is known that KNN algorithm performs reasonably well in case of classification problems with low dimensionality in vector space. Since identification of malicious content in network packets is inherently a classification problem, it can be understood as why KNN performs better than other algorithms. In the following section we have compared

performance while using different parameters and conditions to which algorithms are exposed.

When accuracy is taken into account, in both studies KNN is the more accurate algorithm, while the K-Means is the not accurate.

Table 2: According to Machine Learning Algorithms features-selection.

Algorithm	Features
KNN Algorithm	Bwd Packet Length Std, Flow Bytes/s, Total Length of Fwd Packets, Fwd Packet Length Std, Flow IAT Std, Flow IAT Min, Fwd IAT Total , Flow Duration, Total Backward Packets, Total Length of Fwd Packets, Total Length of Bwd Packets, Fwd Packet Length Max, Fwd Packet Length Mean, Bwd Packet Length Mean.
K-Means Algorithm	

Table 3: Prediction at different values of neighbour sizes.

Algorithm	Neighbours	Prediction
KNN	15	0.99293450574
KNN	10	0.99293450575
KNN	5	0.99463996987

Table 4: Prediction at different values of cluster sizes.

Algorithm	Cluster size	Prediction
K-Means	5	0.3153115244
K-Means	2	0.4683603180
K-Means	2	0.4683603180

Table 5: Application of the features obtained in the first approach.

Machine Learning Algorithm	Evaluation Criteria				
	F-score	Precision	Recall	Accuracy	Time(s)
K Nearest Neighbours	**0.96**	0.96	0.97	0.97	1967.054

As evident from above observations, KNN algorithm proves to be superior in comparison to K-Means algorithm. Results indicate that prediction accuracy reaches as high as 99% in case of KNN algorithm, while K-Means algorithm performs merely as high as 47% (approximately) in best case. So it is established that based on these results KNN algorithm would serve as a better classifier for detecting network based packet anomalies indicating potential threats. Furthermore, a dense forest can be added to predict which kind of potential threat is being detected by the model.

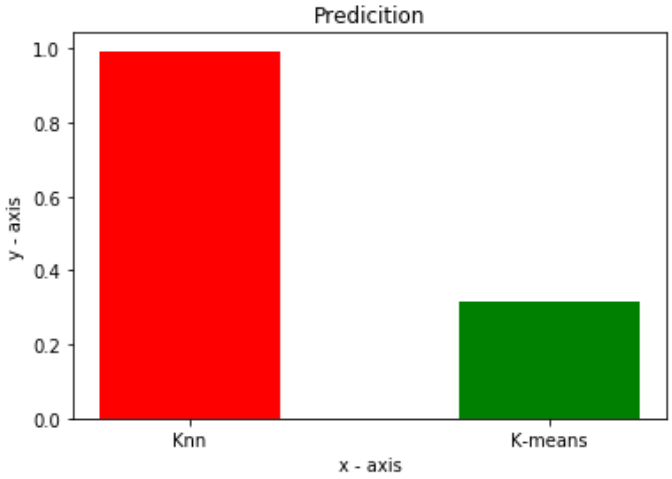

Figure 1: Prediction at Neighbour=15 and cluster size=5

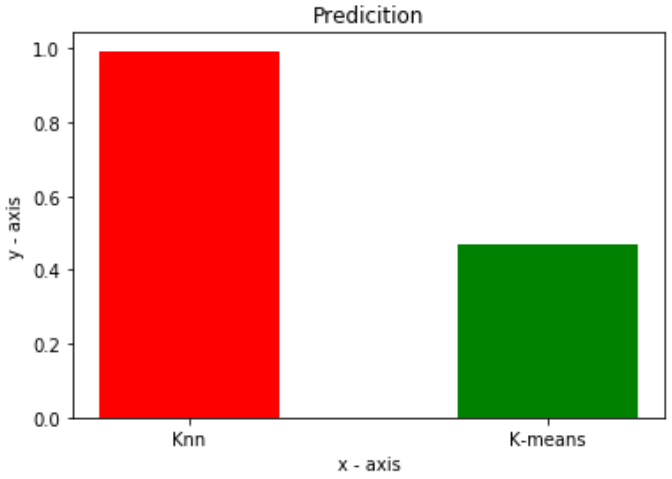

Figure 2: Prediction at Neighbour=10 and cluster size=2

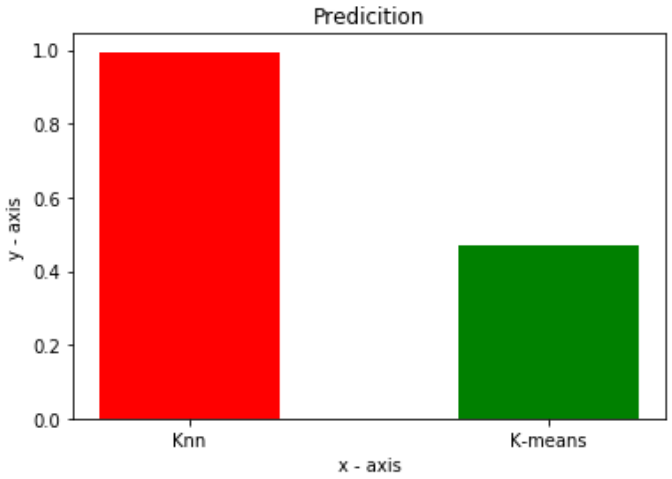

Figure 3: Prediction at Neighbour=5 and cluster size=2

IV. CONCLUSION

In this paper we have discussed about an Intrusion detection system-based machine learning approach with CICIDS 2017 dataset to prevent zero-day attacks. New exploits & attacks are evolving day by day therefore, it becomes more crucial for machines to be secured using advanced intrusion detection systems with the aim to detect intrusions and attack more accurately. Many methods, frameworks are being developed to secure systems from malicious attacks. This method aim to

develop a security system that monitors hosts on a network and analyse the traffic for possible hostile attacks.

This method uses the most well-received public dataset for IDS training, testing, CICIDS, 2017 dataset. This dataset includes the activities of benign and malicious attacks which depicts real time network traffic. The use of a static dataset also raises some issues and concerns as unpatched attacks are emerging every day, existing patterns will no longer be sufficient to detect zero day attacks, dearth of up-to-date labelled data is a barricade in advancement of security systems, another relevant issue is low detection efficiency which can lead to high false positive rate but these can be reduced by selecting only significant features for attack detection process.

This IDS system is based on supervised learning based technique (K-nearest neighbour) which detect intrusion by labelled data. The KNN typically is a non-parametric classifier applied in machine learning. Feature selection plays a very important role in enhancing the detection rate of machine learning technique, reducing false alarms.

V. FUTURE SCOPE

We propose further improvements and research opportunities following from this research.

- Developing IDSs capable of overcoming the evasion techniques remains a major challenge for this area of research. Methods such as fragmentation, encryption, flooding pose a challenge for current IDS as they bypass existing mechanisms.
- Input packet structure covers all the layers of network stack including application layer. Thus, writing application specific intrusion detection is further extension to this research.
- Exploring other techniques for intrusion detection such as neural networks, probabilistic logic, etc.
- It is not always possible to come up with a signature to detect unknown attacks with false positive rates that are low enough for practical use. Therefore combining both signature-based detection and anomaly-based detection to leverage signature based rules without missing unknown attack.

REFERENCES

[1] S. Nerella and M. Shashi, Intrusion Detection Analytics:A Cmprehensive Survey, International Journal of Advanced Scient ific Research and Management , Volume 4 Issue 6, June 2019

[2] P. Gayatri, N.R. Priyanka, N. Nishanth, R. Abhishek and K.A. Vani, Comprehensive Comparative Study on Intrusion Detection System in Cloud Computing, International Journal for Research in Applied Science & Engineering Technology (IJRASET), Volume 3 Issue VI, June 2015.

[3] B.N. Kumar, M.S.V.S. Raju and B.V. Vardhan, A Comparative Survey on the Influence of Machine Learning Techniques on Intrusion Detection System (IDS), IOSR Journal of Engineering (IOSRJEN), Vol. 08, Issue 8 (August. 2018).

[4] Iman Sharafaldin, Arash Habibi Lashkari, and Ali A. Ghorbani, “Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization”, 4th International Conference on Information Systems Security and Privacy (ICISSP), Portugal, January 2018

[5] Arash Habibi Lashkari, Gerard Draper-Gil, Mohammad Saiful Islam Mamun and Ali A. Ghorbani, "Characterization of Tor Traffic Using Time Based Features", In the proceeding of the 3rd International Conference on Information System Security and Privacy, SCITEPRESS, Porto, Portugal, 2017.

[6] Gerard Drapper Gil, Arash Habibi Lashkari, Mohammad Mamun, Ali 3. A. Ghorbani, "Characterization of Encrypted and VPN Traffic Using Time-Related Features", In Proceedings of the 2nd International Conference on Information Systems Security and Privacy(ICISSP 2016) , pages 407-414, Rome , Italy

[7] K. Kostas, "Anomaly Detection in Networks Using Machine Learning," Research Proposal, 23 Mar 2018, 2018.

[8] K. Leung and C. Leckie, "Unsupervised anomaly detection in network intrusion detection using clusters," in Proceedings of the Twenty-eighth Australasian conference on Computer Science-Volume 38, 2005, pp. 333-342: Australian Computer Society, Inc.

[9] I. Sharafaldin, A. Gharib, A. H. Lashkari, and A. A. Ghorbani, "Towards a reliable intrusion detection benchmark dataset," Software Networking, vol. 1, no. 1, pp. 177- 200, 2017